

06-23-00

EK 483548 439US  
June 22, 2000 A

## IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Docket No. AUS000213US1

Assistant Commissioner for Patents  
Washington, D.C. 20231

Sir:

Transmitted herewith for filing is the patent application of Inventor(s):

James Arthur Brewer

Sanjay Gupta

Nicole Dawn Hartman

Mark Alan Jacks

For: AN APPARATUS AND A METHOD FOR DIAGNOSING PROBLEMS ON A  
NETWORK COMPUTER

Enclosed are also:

- ☒ 22 Pages of Specification including an Abstract  
☒ 6 Pages of Claims  
☒ 8 Sheet(s) of Drawings  
☒ A Declaration and Power of Attorney  
☒ Form PTO 1595 and assignment of the invention to IBM Corporation

## CLAIMS AS FILED

FOR	Number Filed		Number Extra		Rate		Basic Fee (\$690)
Total Claims	22	-20 =	2	X	\$ 18	=	\$36
Independent Claims	3	-3 =	0	X	\$ 78	=	\$0
Multiple Dependent Claims	0			X	\$260	=	\$0
Total Filing Fee =							\$726

- ☒ Please charge \$726.00 to IBM Corporation, Deposit Account No. 09-0447.  
☒ The Commissioner is hereby authorized to charge payment of the following fees associated with the communication or credit any over payment to IBM Corporation, Deposit Account No. 09-0447. A duplicate copy of this sheet is enclosed.  
☒ Any additional filing fees required under 37CFR § 1.16.  
☒ Any patent application processing fees under 37CFR § 1.17.

Respectfully,

Casimer K. Salys

Reg. No. 28,900

Intellectual Property Law Dept.

IBM Corporation

11400 Burnet Road 4054

Austin, Texas 75758

Telephone: (512) 823-0092

06/22/00  
Jc796 U.S. PTOPTO  
06/22/00  
081665/60  
Jc796 U.S. PTO

**AN APPARATUS AND A METHOD FOR DIAGNOSING PROBLEMS ON A  
NETWORK COMPUTER**

5                   **BACKGROUND OF THE INVENTION**

**1.    Technical Field**

          The present invention relates generally to an improved network computer and a system for diagnosing  
10 errors. Still more particularly, the present invention provides an apparatus and a method for automatically performing hardware and software diagnostics on a network computer.

15                   **2.    Description of the Related Art**

          With the introduction of the personal computer there was a shift away from centralized, mainframe computing. It was quickly realized that personal computers had to be able to communicate and share resources; this resulted in  
20 the emergence of a variety of network technologies. Even with these technological advances, personal computing focused on the individual user rather than the business organization. The network computer (NC) evolved to help move the focus away from the individual and back to the  
25 organization. This has resulted in an important shift in the way computers are perceived and used within an organization.

          Network computers are the conduits by which corporate resources are accessed. This shift in  
30 perspective from the individual to the organization affects the way that information is created, stored, and disseminated. The end result is greater online

Docket No. AUS000213US1

collaboration, coordination, and communication.

Network computers are configured differently than personal computers with an emphasis on shared resources, including data storage, as contrasted with individual  
5 resources, such as hard drives or floppy drives for local storage on a personal computer. As a minimum, a network computer contains a monitor, keyboard, mouse, network interface, and audio circuitry. Software support includes the Java virtual machine and runtime  
10 environment, the Java API class libraries, TCP/IP networking support, a web browser, email support, and multimedia support. The use of the Java Virtual Machine and Java Runtime Environment greatly enhances application security. In addition, network server security controls  
15 can be used to limit the access of individual users. As one of ordinary skill in the art will appreciate, although network computers commonly use Java as the language of choice, development of software for network computers could use a wide variety of programming  
20 environments.

The NC environment is based on centralized system software, thus lowering overall computing costs. This lower total cost of ownership is a primary consideration for organizations that deploy NCs. Because all software  
25 is installed and maintained on network servers, large organizations do not require the huge support staffs needed by those that deploy personal computers and workstations.

Despite the simplicity of the network computer,  
30 problems can occur both in hardware and software. Therefore, it would be advantageous to have a diagnostic apparatus and method to quickly determine the source of

,

,

Docket No. AUS000213US1

### SUMMARY OF THE INVENTION

An apparatus and a method is presented for  
5 diagnosing hardware and software problems on a network  
computer through the use of a diagnostic adapter card.  
The diagnostic adapter card is installed in an open slot  
in the system bus and thus has access to all the network  
computer electronics and peripherals. Diagnostic  
10 programs are run and the results are logged. By  
analyzing these results, the source of the problem can be  
determined and reported for service or repair.

The diagnostic adapter card contains its own  
processor, read only memory, random access memory, and  
15 interface logic for connecting to the system bus, a set  
of external wrap cables for peripheral connector testing,  
and an external reporting device. Some tests require use  
of a wrap cable to connect two or more ports together.  
Other tests, such as PCI bus timing or memory integrity,  
20 do not require the use of wrap cables.

Depending on the nature of the problem and assuming  
the network connection is functional, tests may be run  
remotely by a system administrator. For some tests,  
local assistance may be required to install wrap cables.  
25 The diagnostic adapter card can also be used to monitor  
system performance over longer periods of time and help  
in the detection of intermittent system or software  
problems.

**BRIEF DESCRIPTION OF THE DRAWINGS**

The novel features believed characteristic of the invention are set forth in the appended claims. The invention itself, however, as well as a preferred mode of use, further objectives and advantages thereof, will best be understood by reference to the following detailed description of an illustrative embodiment when read in conjunction with the accompanying drawings, wherein:

**Figure 1** is a pictorial representation of a distributed data processing system in which the present invention may be implemented;

**Figure 2** is a block diagram of a network computer in which the present invention may be implemented;

**Figure 3** is a block diagram for the diagnostic adapter card in accordance with a preferred embodiment of the present invention;

**Figure 4A** shows the use of a serial port wrap cable in accordance with a preferred embodiment of the present invention;

**Figure 4B** gives a flowchart for the serial port test in accordance with a preferred embodiment of the present invention;

**Figure 5A** shows the use of an audio port wrap cable in accordance with a preferred embodiment of the present invention;

**Figure 5B** gives a flowchart for the audio test in accordance with a preferred embodiment of the present invention;

**Figure 6A** shows the use of a multi-drop wrap cable between the network port, the external network

Docket No. AUS000213US1

connection, and the diagnostic port in accordance with a preferred embodiment of the present invention;

**Figure 6B** gives a flowchart for the network circuitry test in accordance with a preferred embodiment  
5 of the present invention;

**Figure 7A** shows the use of a wrap cable between a USB (Universal Serial Bus) port and a parallel port in accordance with a preferred embodiment of the present invention;

10 **Figure 7B** gives a flowchart for the USB to parallel port test in accordance with a preferred embodiment of the present invention;

**Figure 8** gives a flowchart for a test of system memory in accordance with a preferred embodiment of the  
15 present invention; and

**Figure 9** gives a flowchart for testing devices connected to the PCI bus in accordance with a preferred embodiment of the present invention.

**DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT**

With reference now to the figures, and in particular  
5 with reference to **Figure 1**, a pictorial representation of  
a distributed data processing system is depicted in which  
the present invention may be implemented.

Distributed data processing system **100** is a network  
of computers. Distributed data processing system **100**  
10 contains network **102**, which is the medium used to provide  
communications links between various devices and  
computers connected within distributed data processing  
system **100**. Network **102** may include permanent  
connections, such as wire or fiber optic cables, or  
15 temporary connections made through telephone connections.

In the depicted example, servers **104**, **114**, **116** and  
**118** are connected to network **102**. Storage units **106** and  
**122** are also connected to network **102**, providing backup  
support for any or all of servers **104**, **114**, **116** and **118**.  
20 Storage unit **122** provides dedicated backup support for  
server **104**. In addition, network computers **108**, **110** and  
**112** are also connected to network **102**. For the purposes  
of this application, a network computer is any computer  
with a processor and boot code that is coupled to a  
25 network to receive its operating system and application  
software from the network. Distributed data processing  
system **100** may include additional servers, clients, and  
other devices not shown.

In the depicted example, servers **104**, **114**, **116** and  
30 **118** provide storage for data from network computers **108**,  
**110** and **112**. These four servers also provide data, such



Docket No. AUS000213US1

as boot files, operating system images, and applications to network computers **108**, **110** and **112**. Network computers **108**, **110** and **112** are clients to one or all of servers **104**, **114**, **116** and **118**. Support for a particular application being performed on one of network computers **108**, **110** and **112** may be by one of servers **104**, **114**, **116** and **118**. Additionally servers **104**, **114**, **116** and **118** may provide backup support for each other. In the event of a server failure, a redundant backup server may be allocated by the network administrator, in which case requests directed to the failed server are routed to the redundant backup server.

In a similar manner, storage units **106** and **122** provide data backup support for servers **104**, **114**, **116** and **118**. However, rather than the network administrator allocating a data backup storage unit at each use, data backup allocation is set, and data backup transfer occurs at low usage times, typically after midnight, between any of servers **104**, **114**, **116** and **118** and storage units **106** and **122**.

In the depicted example, distributed data processing system **100** may be the Internet, with network **102** representing a worldwide collection of networks and gateways that use the TCP/IP suite of protocols to communicate with one another. At the heart of the Internet is a backbone of high-speed data communication lines between major nodes or host computers consisting of thousands of commercial, government, education, and other computer systems that route data and messages. Of course, distributed data processing system **100** also may be implemented as a number of different types of networks, such as, for example, an intranet or a local

Docket No. AUS000213US1

area network.

**Figure 1** is intended as an example and not as an architectural limitation for the processes of the present invention.

5        With reference now to **Figure 2**, a block diagram of a data processing system in which the present invention may be implemented is illustrated. Data processing system **200** is an example of a network computer. Data processing system **200** employs a peripheral component interconnect  
10        (PCI) local bus architecture and an ISA (Industry Standard Architecture) bus architecture. Processor **202** connects to PCI bridge **208** and L2 Cache **209** through processor bus **201**. DRAM memory **204** is connected to PCI bridge **208** through memory bus **203**. PCI bridge **208** may  
15        also include an integrated memory controller for processor **202**.

Additional connections to PCI local bus **206** may be made through direct component interconnection or through add-in boards. In the depicted example, local area  
20        network (LAN) controller **210** connects to network **207**. Video controller **218**, assisted by frame buffer **219**, controls video terminal **205**. PCI Riser Card **213** contains two additional PCI slots. In this example, diagnostic adapter card **222** with associated Readout **212** are  
25        connected to one of these card slots.

PCI Bridge **208** connects to ISA Bridge **214**, which is used to connect other I/O ports, including USB ports 1 and 2 **215**. Audio control **216** connects through ISA bus **211**. The audio control connects to microphone **217** and  
30        headphone **220**. As anyone skilled in the art will appreciate, a speaker could be added to the audio

Docket No. AUS000213US1

circuitry.

ISA bridge **214** connects via ISA bus **211** to System Boot Flash **221**, Real Time Clock **223**, and "Super I/O" **224**. This "Super I/O" includes a variety of external ports:  
5 Serial 1 **225**, Serial 2 **226**, Parallel **227**, IDE **228**, Compact Flash **229**, Keyboard **230**, and Mouse **231**.

An operating system runs on processor **202** and is used to coordinate and provide control of various components within network computer **200** in **Figure 2**. In a  
10 network computer, the Java runtime environment and the Java virtual engine perform the functions commonly found in an operating system.

Those of ordinary skill in the art will appreciate that the hardware in **Figure 2** may vary depending on the  
15 implementation. For example, other peripheral devices may be used in addition to or in place of the hardware depicted in **Figure 2**. The depicted example is not meant to imply architectural limitations with respect to the present invention. For example, the processes of the  
20 present invention may be applied to multiprocessor data processing systems.

Problems in a network computer environment might be caused by a hardware failure of a particular component, a software failure in the operating system, a software  
25 failure in a particular application, or a combination of these factors. It is important to identify and locate the problem quickly in order to minimize user frustration and lower maintenance costs.

The network station diagnostic adapter card plugs  
30 into the PCI bus so that it can access the electronics of the network computer. The adapter contains its own programming and electronics so that it can perform

Docket No. AUS000213US1

diagnostic functions even when there are hardware or software failures. Once installed, the diagnostic adapter card can also be left in the machine to collect data when the network computer is functioning properly.

- 5 If problems are detected, the network administrator can be notified via the network adapter **207**, if it is functional, via the video display **205**, if it is functional, or via a set of diagnostic indicators **212** visible externally if the network interface and display  
10 are not functioning properly.

The diagnostic adapter card can perform a variety of hardware and software diagnostics. The PCI bus can be checked for timing and control problems, addressing problems, and data problems. In particular, direct memory  
15 access (DMA) and bus mastering operations can be performed. The system memory can be checked. Watchdog or timeout functions can be tested. The various input/output ports (serial port, parallel port, USB port, audio port, etc.) can be checked for integrity using data  
20 wrap cables when appropriate. The System Boot FLASH memory **221** can be verified and, if necessary, restored. As one of ordinary skill in the art will appreciate, this diagnostic testing could be adapted to a particular hardware and software environment.

- 25 With reference now to **Figure 3**, a block diagram is given for the diagnostic adapter card **222**. The various components on diagnostic adapter card **222** communicate via Memory Controller North Bridge **302**. In particular, the boot code is shadowed from Read Only Memory **304** into  
30 Random Access Memory (RAM) **306**. In this way the boot code can be executed from RAM **306** without paying the performance penalty of booting directly from ROM **304**.

Docket No. AUS000213US1

Data storage and retrieval uses Random Access Memory **306** under the control of Processor **308**. By having memory and processing capabilities on the diagnostic adapter card, some diagnostic tests can often be performed even when  
5 the memory and/or processor in the network computer are defective. Simple diagnostic tests can be stored in ROM **304** but more complex diagnostic tests will be loaded in RAM **306** from the network, from flash memory, or from some other data source.

10 The diagnostic adapter card sends signals to various devices in the computer via the PCI bus. In particular, Memory Controller North Bridge **302** sends and receives signals from Memory Controller South Bridge **310**, which, in turn, communicates these signals using the PCI bus.  
15 Card edge tab **312** plugs into PCI riser **213**.

Connectors are provided on the back of the diagnostic adapter card for two primary purposes: communicating test results and providing connections for certain types of wrap testing. Data can be transferred  
20 to or from reporting devices via connection **316**, which interfaces through miscellaneous interface logic **314**. This interface logic communicates with other components on the card via Memory Controller North Bridge **302**. The reporting device may be a set of LED indicators, an LCD  
25 display, LED digital alphanumeric display, or any similar technology. It should be noted that results could also be reported via the a network adapter, such as network controller **210** in **Figure 2**, if that portion of the network computer is working correctly. Results may  
30 also be reported via connector **318** along with use of an appropriate Network multi-drop wrap cable, if the

Docket No. AUS000213US1

diagnostic adapter card has network capabilities.

Some forms of testing require connections with wrap cables. For example, to test a network port, a multi-drop wrap cable needs to be installed connecting  
5 the network port to the external network connection and to the diagnostic adapter card. Connector **318**, which is connected to interface logic **314**, provides a way to monitor or connect to the network from the diagnostic adapter card via a multi-drop wrap cable. Using this  
10 multi-drop cable, the diagnostic adapter card can monitor and compare data sent out of or in to the network port with the external network connection. Additionally, even if the external network connection is not available, the diagnostic adapter card can communicate directly with the  
15 network port. If the external network connection is available, the diagnostic adapter card can communicate test results directly to a recipient connected to the external network.

To test the network port operation, a sequence of  
20 data can be sent to the network port via either the network server or the wrap cable. The network port can respond by sending a signal back to the network server or to the diagnostics card through the wrap cable or through the system PCI bus, depending on the circumstance. One  
25 particular test for network card integrity will be given in detail in **Figures 6A** and **6B** below.

With reference now to **Figure 4A**, a block diagram shows the use of a serial wrap cable to test the serial ports. It is assumed network computer **402** has a variety  
30 of ports **404**. In particular, it is assumed there are two serial ports, referenced as serial 1 and serial 2. Furthermore, to perform the diagnostic test on these

Docket No. AUS000213US1

ports, wrap cable **406** is used to connect these two ports. Serial ports are capable of both input and output. We will describe a test of sending a signal out of serial 1 and receiving it at serial 2, but, as will be appreciated  
5 by those of ordinary skill in the art, the test could easily be reversed with serial 2 sending a signal to serial 1.

With reference now to **Figure 4B**, a flowchart illustrates the steps in performing the serial port test.  
10 Using the PCI bus, which in this example is assumed to be functional, diagnostic adapter card **300** or system processor **202** transfers test data to the serial 1 circuit (step **408**). This data transfer is via PCI bridge **208** and ISA bridge **214**. The data is stored temporarily in the  
15 serial 1 buffer (step **410**). Using the serial device control register, serial 1 circuit is instructed to output the data on serial 1 port (step **412**). At the same time, serial 2 circuit has been instructed to receive data at serial port 2 (step **414**). Wrap cable **404** is  
20 essential to complete this transfer. The received data is placed in serial 2 buffer (step **416**). This data is then transferred back to the diagnostic adapter card via the PCI bus (step **418**). The received data is compared with the original data (step **420**), results of this  
25 comparison are logged (step **422**), and appropriate success or failure test notification is posted to the diagnostics card readout, to the NC video display, or via the network connection.

For this particular test, 100% agreement in data  
30 would be expected for success and any discrepancy in data values, no matter how small, would represent failure. As

Docket No. AUS000213US1

one of ordinary skill in the art will appreciate, the nature of the comparison and the standard for "success" will be dependent on the device and type of data, as will be seen in the next test of audio data.

5       With reference now to **Figure 5A**, a block diagram shows the use of an audio wrap cable to test the audio circuits. It is assumed network computer **402** has a variety of ports **404**. In particular, it is assumed the audio circuitry has a port for input from a microphone  
10       and a general audio output port. To perform one of the audio diagnostic tests, audio wrap cable **506** is used to connect the audio output port to the microphone port. Unlike the previous example for serial ports, these audio ports are unidirectional and are sending or receiving  
15       analog signals, as contrasted to digital signals.

      With reference now to **Figure 5B**, a flowchart illustrates the steps in performing the audio test for the audio output and microphone ports. Diagnostic card **300** sends a digital file via the PCI bus, the PCI bridge,  
20       and the ISA bridge to the audio circuitry where the digital file is converted, byte-by-byte, into analog data that generates an audio test signal (step **510**). This test signal is sent to the audio output port (step **512**). Audio Wrap cable **506** feeds this analog signal directly  
25       back to the microphone port, where the signal is received as analog input and converted to digital data (step **514**). This digital signal is compared with the expected or original signal (step **516**) and the result of this comparison is sent back to the diagnostic adapter card  
30       for logging (step **518**). The received signal will be analyzed to insure appropriate frequency, phase, and



Docket No. AUS000213US1

voltage levels are being generated and properly received. The audio circuitry is considered to have passed the test only if the audio parameters fall within a predetermined percentage of the original signal; for instance, the  
5 audio input voltage is expected to be between 95% to 105% of the original audio output voltage.

As one of ordinary skill in the art will appreciate, a variety of different audio tests can be performed. Although the audio test described in **Figure 5** may be  
10 initiated remotely, a human user would need to be present to install the wrap cable.

The next test involves the use of connector **318** on diagnostic adapter card **300**. As seen in **Figure 6A**, network computer **402** has a diagnostic adapter card  
15 installed, as evidenced by a diagnostic port among ports **404**. To perform this test, the network data multi-drop wrap cable **606** is connected between the network port, the external network connection, and the diagnostic port. Network traffic that flows between the network port and  
20 the external network connection is monitored by the diagnostic port. A variety of network protocols, such as ethernet or token ring, can be accommodated by a test setup such as this.

With reference now to **Figure 6B**, a flowchart  
25 illustrates the steps in performing the network test. Using the PCI bus, diagnostic adapter card **300** transfers test data to the network interface circuitry (step **610**). The data is then packaged according to the rules of the implemented network and sent to the network port (step  
30 **612**). Wrap cable **606** transfers this data back to the diagnostic adapter card as well as sending it to the network server, if it is connected. The test data is

Docket No. AUS000213US1

received at the diagnostic adapter card (step **614**) and placed in the diagnostic adapter card buffer (step **616**). The received data is compared with the original data (step **618**) and results of this comparison are logged (step **620**). For this particular test, 100% agreement in data would be expected for success and any discrepancy in data values, no matter how small, would represent failure. As one of ordinary skill in the art will appreciate, the signal direction on this test could be reversed with either the diagnostic adapter card or the network server sending the test data on the network and the network circuitry receiving the data and sending it via the PCI bus to the diagnostic adapter card for analysis. This wrap cable also enables the Diagnostic Card to communicate directly with the network server to pass on test results and conclusions.

With reference now to **Figure 7A**, a block diagram shows the use of a wrap cable to test data transfer from a USB (Universal Serial Bus) port to a parallel port. It is assumed network computer **402** has at least one parallel port and one USB port among the variety of ports **404**. To perform this diagnostic test, wrap cable **706** is used to connect these two ports. Since both ports are bi-directional, the data transfer could be tested in either direction. We will describe a test of sending a signal out of USB 2 and receiving it at the parallel port, but, as will be appreciated by those of ordinary skill in the art, the test could easily be reversed with the parallel port sending data to USB 2. What makes this situation different than the serial to serial test described in **Figure 4** is that the wrap cable is "active," in the sense that it converts a bit-serial signal to a

Docket No. AUS000213US1

bit-parallel signal or vice versa depending on the direction of transfer.

With reference now to **Figure 7B**, a flowchart illustrates the steps in performing the USB to parallel test. Using the PCI bus, diagnostic adapter card **300** transfers test data to the USB 2 circuit (step **708**). This data transfer is via PCI bridge **208** and ISA bridge **214**. The data is stored temporarily in the USB 2 buffer (step **710**). Using the USB controller device register, the USB 2 circuit is instructed to output data on the USB 2 port (step **712**). Wrap cable **706** converts this bit-serial signal to a bit-parallel signal (step **714**) before receipt at the parallel port (step **716**). The received data is placed in the parallel received data buffer (step **718**). This data is then transferred back to the diagnostic adapter card via the PCI bus (step **720**). The received data is compared with the original data and results of this comparison are logged (step **722**). For the test to be successful, the received parallel data must compare "exactly" to the USB transmitted data.

Not all tests performed by the diagnostic adapter card require the temporary installation of wrap cables. **Figure 8** gives a flowchart for one such test, a test of system memory. The memory test data is loaded into the diagnostic adapter card memory (step **812**). Using bus mastering, the diagnostic adapter card carries out the memory test (step **814**) as a sequence of memory write and read commands to the NC system memory. Performing a memory test requires access to Memory Bus **203** via PCI bridge **208**. The results of the memory test are logged (step **816**) and success determined based on whether the

Docket No. AUS000213US1

data read back from memory exactly compared to the data written to memory.

This flowchart is deceptively simple since it does not specify the details of the memory test performed. As  
 5 one of ordinary skill in the art will appreciate, there are a wide variety of memory test programs. Some of the common tests are a "walking ones" address pattern, the "Modulo-X" algorithm where X is an appropriately selected small integer, the "own address" test, the "walking ones"  
 10 with inversion, and so forth. The diagnostic adapter card will contain tests appropriate to detect both "hard" memory errors and intermittent or "soft" memory errors.

With reference now to **Figure 9**, a flowchart illustrates the steps in performing a test of devices  
 15 connected to the PCI bus. Although the steps discuss devices connected to a PCI bus, a similar test could be designed for other bus architectures. This test has three major components: gathering information about devices connected to the bus, comparing this information  
 20 with a list of known devices, and then testing the control registers and data registers for each device.

The first step is to discover what devices are currently attached to the bus. If there are no other devices to be discovered (step **910**: no), then control  
 25 transfers to step **914** to continue testing each device individually. If there are more devices on the bus (step **910**: yes), then the next vendor ID and device ID are read and added to the list of devices (step **912**). Control transfers to step **910** and this process continues  
 30 until all devices have been discovered.

The list of discovered devices is compared with a list of known devices (step **914**). Any discrepancies

Docket No. AUS000213US1

between the two lists are logged (step **916**). Devices connected to a PCI bus usually contain control registers, status registers, and data registers. Typically control and data registers can be written to and read from. The  
5 next stage of testing is to see if these registers are accessible and can be changed.

If there are more devices to be tested (step **918**: yes), then the diagnostic adapter card will write a known pattern of data to a device register via the PCI bus  
10 (step **920**). These patterns will be carefully chosen so that they do not disrupt the device or cause undesirable behavior. The device register will be read to verify the register is readable and writeable (step **922**). The results are logged for each register tested (step **924**).  
15 When all the devices are tested (step **918**: No), the PCI bus test is completed and results may be presented via any one of the output mechanisms supported by the diagnostics adapter.

**Figures 4-9** have presented a sampling of tests and  
20 do not imply limitations to only tests of these types. Nor do they imply the order in which the tests are performed. For example, the test of communications with devices on the PCI bus given in **Figure 9** would normally be performed before other tests that assume a functional  
25 PCI bus, such as tests given in **Figures 4-8**. Other tests may log data during normal operation of the network computer over an extended period of time. It should also be obvious that test programs can be loaded into the diagnostic adapter card via either the NC network  
30 connection and PCI bus or via a direct network connection using the multi-drop wrap cable. By using the diagnostic adapter card, the nature of the hardware and/or software

Docket No. AUS000213US1

problems can be quickly and accurately identified, thus minimizing the system downtime and reducing the overall maintenance costs.

The description of the present invention has been  
5 presented for purposes of illustration and description  
but is not intended to be exhaustive or limited to the  
invention in the form disclosed. Many modifications and  
variations will be apparent to those of ordinary skill in  
the art. The embodiment was chosen and described in  
10 order to best explain the principles of the invention and  
the practical application, and to enable others of  
ordinary skill in the art to understand the invention for  
various embodiments with various modifications as are  
suited to the particular use contemplated.

Docket No. AUS000213US1

**CLAIMS:**

What is claimed:

1. A method in a network computer for diagnosing a  
5 problem, the method comprising the steps of:  
    running diagnostic programs on a diagnostic adapter  
    card coupled to the network computer;  
    reporting the results from running the diagnostic  
10 programs; and  
    analyzing the results from running the diagnostic  
    programs to determine a cause of the problem.
2. The method of claim 1, wherein the diagnostic  
15 adapter card is coupled to the network computer by way of  
    an open slot on a PCI (Peripheral Component Interconnect)  
    bus in the network computer.
3. The method of claim 1, wherein running diagnostic  
20 programs includes running a program to test bus timing,  
    to test bus mastering, to test direct memory access  
    operations, to test data and control registers associated  
    with devices connected to a system bus, to test system  
    memory, to test timeout functions, to test a boot flash  
25 monitor, to test input/output integrity for one or more  
    devices selected from a keyboard, a mouse, a graphics  
    adapter, a serial port, a parallel port, a universal  
    serial bus port, a microphone, a speaker, and an audio  
    output port.
- 30 4. The method of claim 1, wherein reporting results  
    includes sending data to a remote workstation, storing  
    information in a log file, and displaying a result code

Docket No. AUS000213US1

on a display device connected to the diagnostic adapter card.

5. The method of claim 1, wherein analyzing results  
5 includes detecting the absence of a response from a component, detecting a discrepancy between an expected response and an actual response, and recognizing errors in signal timing.
- 10 6. The method of claim 1, wherein the cause of the problem includes detecting a nonfunctioning component, detecting an intermittently failing component, and detecting a faulty software program.
- 15 7. An apparatus in a network computer for diagnosing a problem, the apparatus comprising:  
a processing means for executing diagnostic programs on the diagnostic adapter card;  
a reporting means for reporting results from  
20 executing the diagnostic programs;  
an analyzing means for analyzing the results from executing the diagnostic programs to determine a cause of the problem.
- 25 8. The apparatus of claim 7, wherein the apparatus comprises a diagnostic adapter card installed in an open slot on a PCI (Peripheral Component Interconnect) bus in the network computer and one or more wrap cables.
- 30 9. The diagnostic adapter card of claim 8, wherein the diagnostic adapter card includes a processing means to execute the diagnostic programs, a read only memory to



Docket No. AUS000213US1

boot the processing means, a random access memory to store diagnostic programs and data to be processed by the processing means, a first external connector to interface with a reporting device, and a second external connector  
5 to connect a wrap cable to send or receive sample data.

10. The diagnostic adapter card of claim 8, wherein an integrity of a first input/output port in the network computer and a second input/output port in the network  
10 computer is tested by connecting a wrap cable between the first input/output port and the second input/output port.

11. The diagnostic adapter card of claim 10, wherein the diagnostic adapter card processing means sends a first  
15 set of data on a system bus to a device card for the first input/output port, the first set of data is sent via the wrap cable to the second input/output port, a device card for the second input/output port receives a signal as a second set of data, the device card for the  
20 second input/output port sends the second set of data back to the diagnostic adapter card on the system bus, and the processing means on diagnostic adapter card compares the first set of data with the second set of data to determine any errors.

25

12. The diagnostic adapter card of claim 10, wherein the wrap cable between the first input/output port and the second input/output port converts a format of the data without changing content of the data.

30

13. The diagnostic adapter card of claim 10, wherein the processing means for executing diagnostic programs

Docket No. AUS000213US1

includes executing a program to test one of bus timing, bus mastering, direct memory access operations, data and control registers associated with devices connected to the system bus, system memory, timeout functions, system processor sequencing, a boot flash monitor, and input/output integrity for one or more devices selected from a keyboard, a mouse, a graphics adapter, a serial port, a parallel port, a universal serial bus port, a microphone, a speaker, and an audio output port.

10

14. The diagnostic adapter card of claim 10, wherein the reporting means for reporting results includes one of sending data to a remote workstation, storing information in a log file, and displaying data on a display device connected to the diagnostic adapter card.

15

15. The diagnostic adapter card of claim 10, wherein the analyzing means for analyzing results includes one of recognizing known error codes, detecting the absence of a response from a component, and recognizing errors in signal timing.

20

16. The diagnostic adapter card of claim 10, wherein the cause of the problem includes one of detecting a nonfunctional component, detecting an intermittent component, and detecting a faulty software program.

25

17. An computer program product for diagnosing a problem, the computer program product comprising:

30

instructions for diagnostic programs on a diagnostic adapter card;

instructions for reporting results from executing

Docket No. AUS000213US1

the diagnostic programs;

instructions for analyzing the results from  
executing the diagnostic programs to determine a cause of  
the problem.

5

18. The computer program product of claim 17, wherein  
instructions for executing diagnostic programs include  
instructions for executing an integrity test of a first  
input/output port and a second input/output port that are  
10 connected by a wrap cable between the first input/output  
port and the second input/output port.

19. The computer program product of claim 17, wherein  
the instructions for executing diagnostic programs  
15 includes executing a program to test one of bus timing,  
bus mastering, direct memory access operations, data and  
control registers associated with devices connected to  
the system bus, system memory, timeout functions, system  
processor sequencing, a boot flash monitor, and  
20 input/output integrity for one or more devices selected  
from a keyboard, a mouse, a graphics adapter, a serial  
port, a parallel port, a universal serial bus port, a  
microphone, a speaker, and an audio output port.

25 20. The computer program product of claim 17, wherein  
the instructions for reporting results includes one of  
sending data to a remote workstation, storing information  
in a log file, and displaying information on a display  
device connected to the diagnostic adapter card.

30

21. The computer program product of claim 17, wherein  
the instructions for analyzing results includes detecting

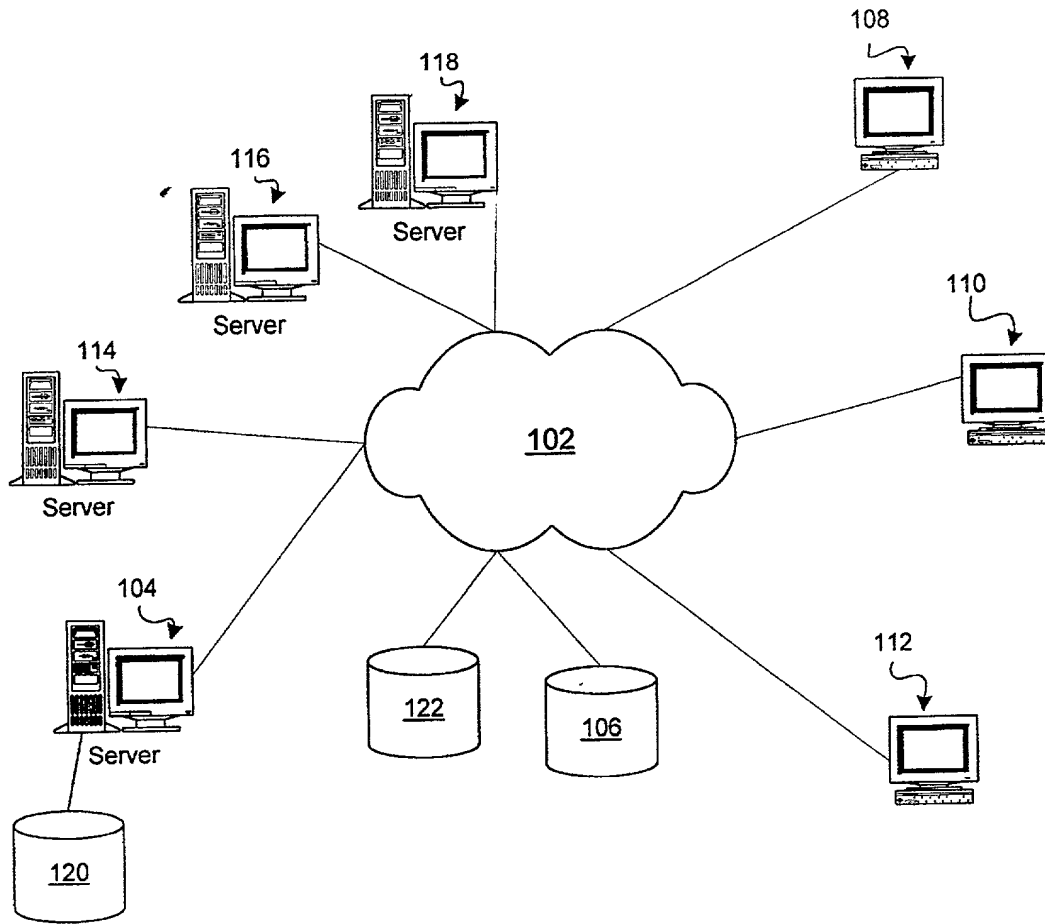
the absence of a response from a component, detecting a discrepancy between an expected response and an actual response, and recognizing errors in signal timing.

22. The computer program product of claim 17, wherein the cause of the problem includes one of detecting a nonfunctional component, detecting an intermittently failing component, and detecting a faulty software program.

**ABSTRACT OF THE DISCLOSURE****AN APPARATUS AND A METHOD FOR DIAGNOSING PROBLEMS ON A  
NETWORK COMPUTER**

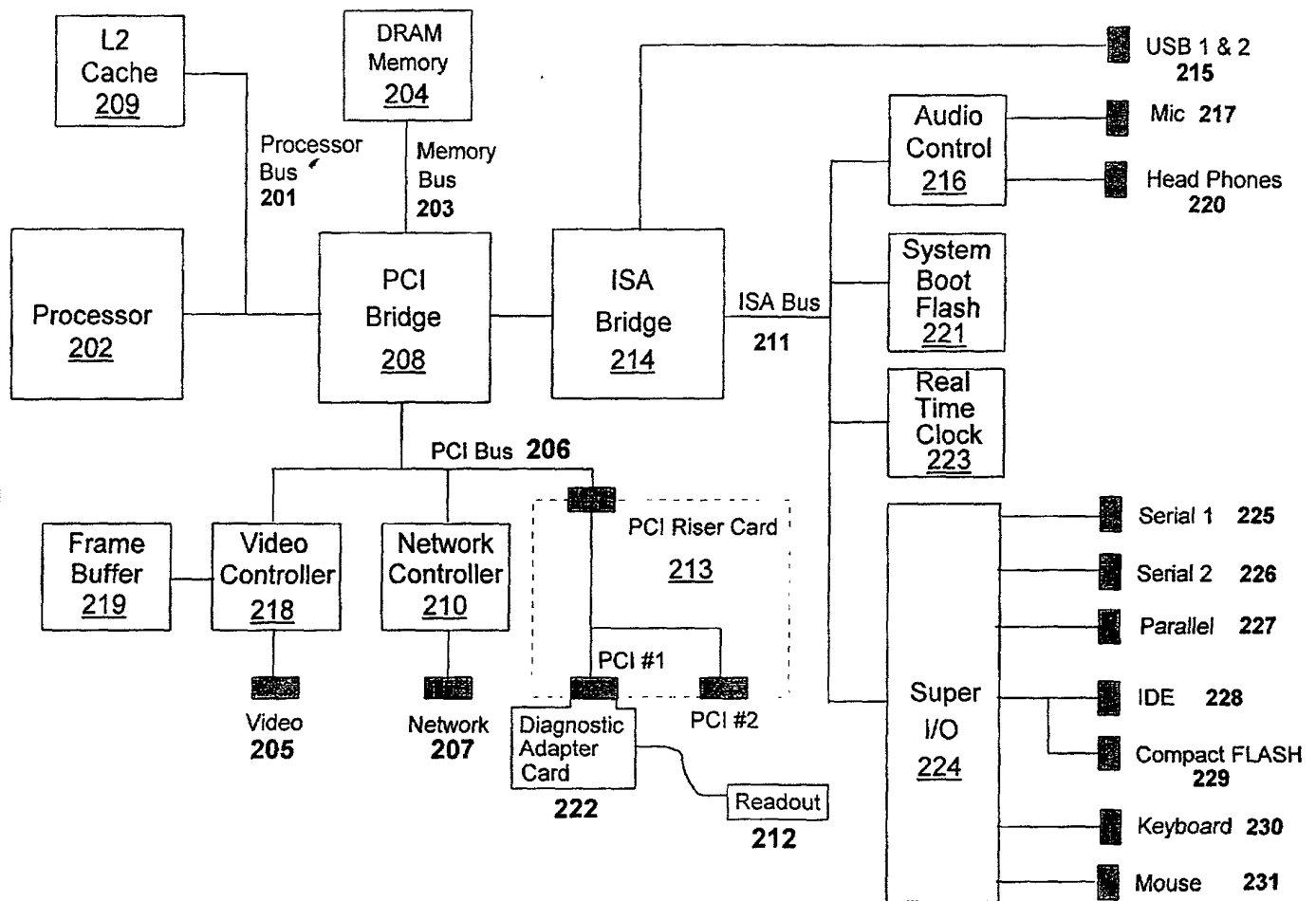
5

An computer program product and a method is presented for diagnosing hardware and software problems on a network computer through the use of a diagnostic adapter card. The diagnostic adapter card is installed  
10 in an open slot in the system bus and thus has access to all the network computer electronics and peripherals. Diagnostic programs are run and the results are logged. By analyzing these results, the source of the problem can be determined and reported for service or repair. The  
15 diagnostic adapter card contains its own processor, read only memory, random access memory, and interface logic for connecting to the system bus, a set of external wrap cables for peripheral connector testing, and an external reporting device. Some tests require use of a wrap cable  
20 to connect two or more ports together. Other tests, such as system bus timing or memory integrity, do not require the use of wrap cables.



100  
Network  
**Figure 1**

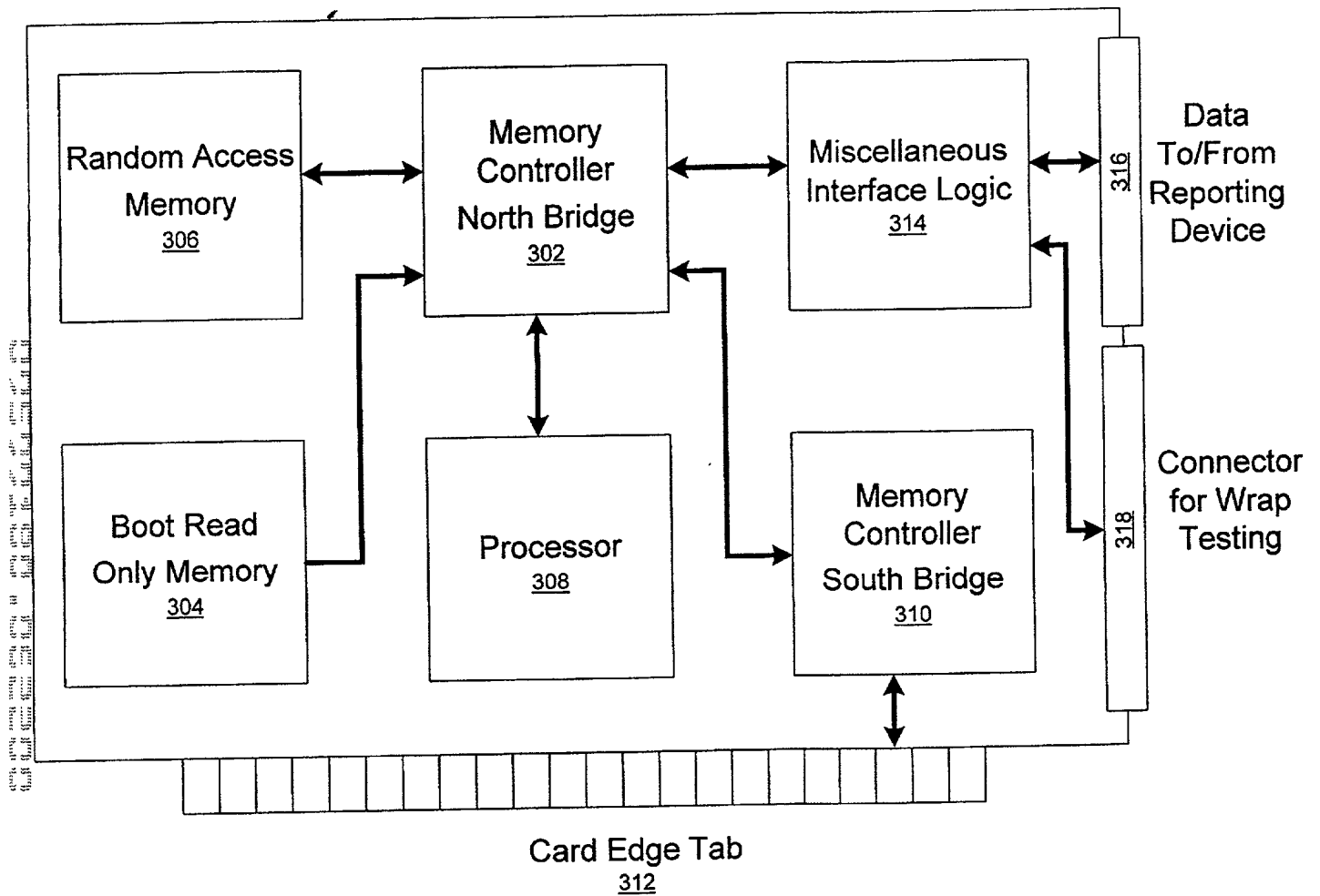
AUS000213US1



200  
Network Computer with  
diagnostic adapter card

**Figure 2**

AUS000213US1



DiagnosticAdapter Card  
222

Figure 3

AUS000213US1



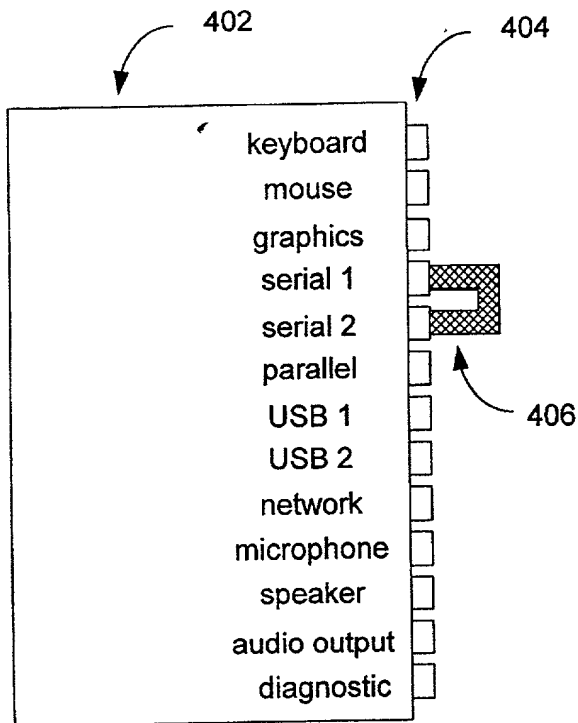


Figure 4A

AUS000213US1

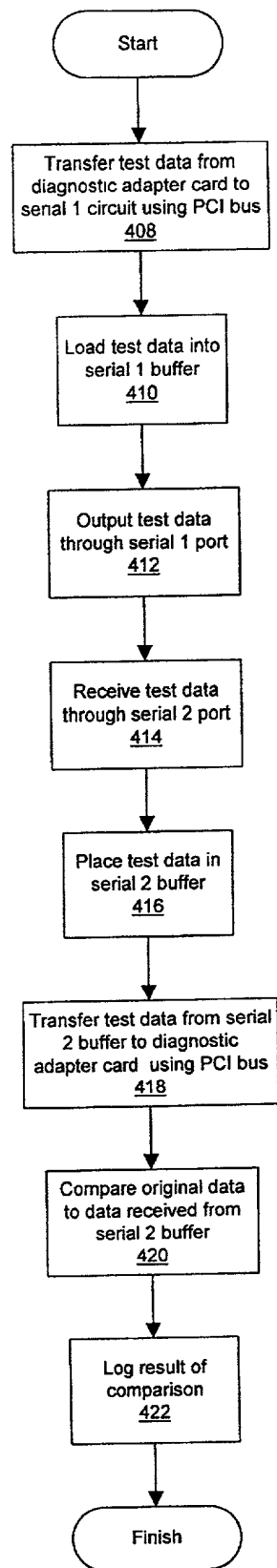


Figure 4B

AUS000213US1

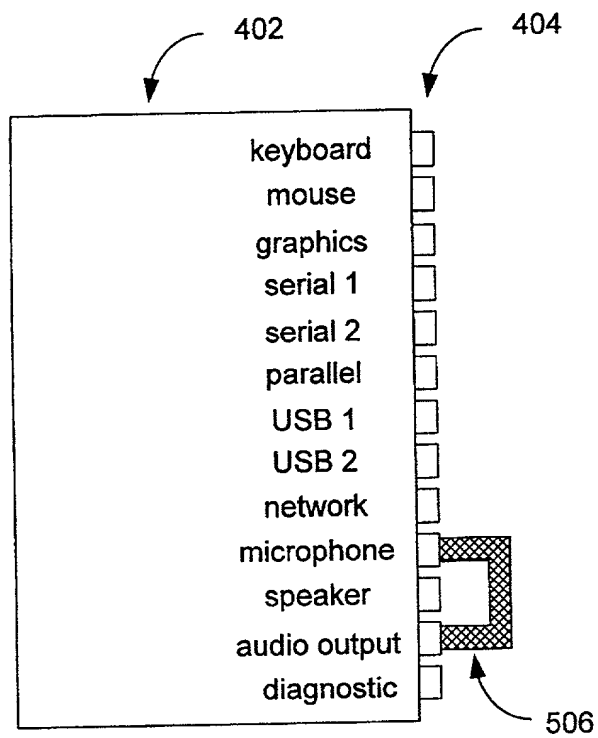


Figure 5A

AUS000213US1

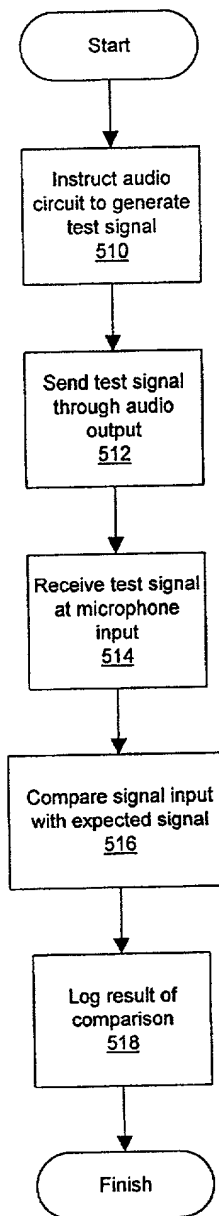


Figure 5B

AUS000213US1

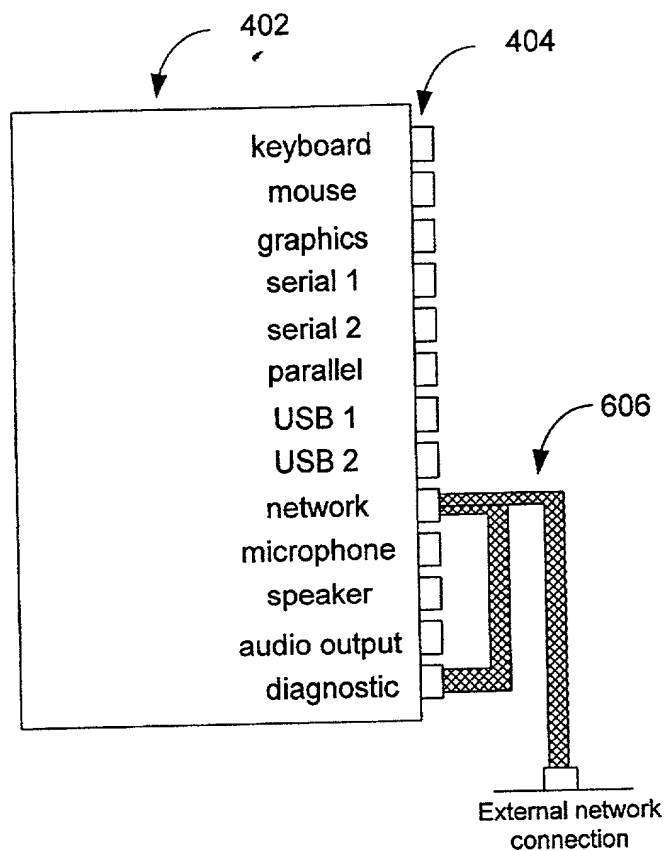


Figure 6A

AUS000213US1

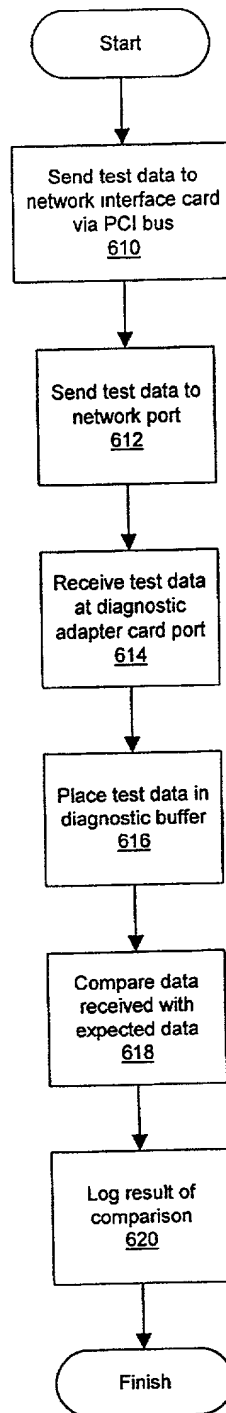
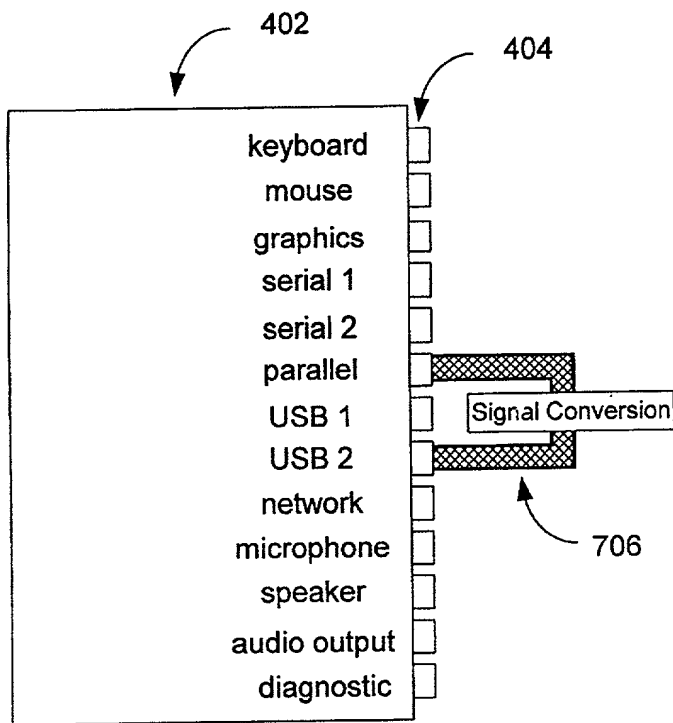
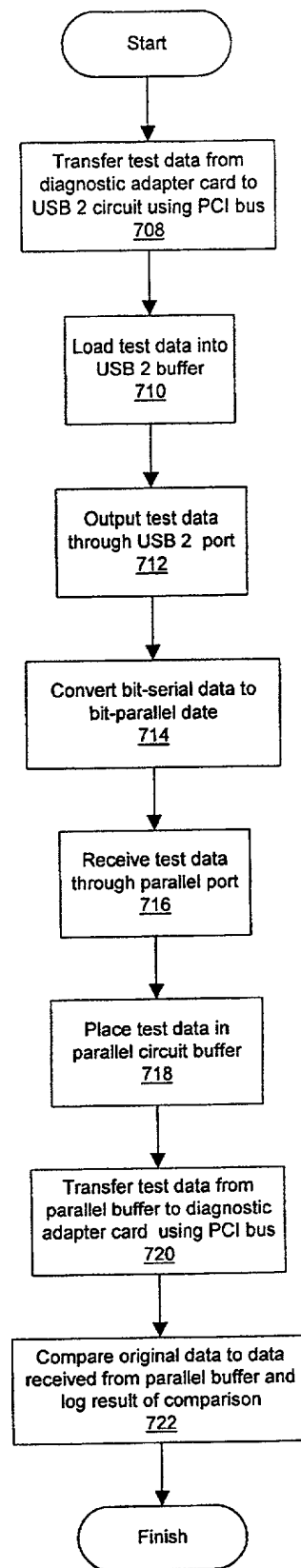


Figure 6B

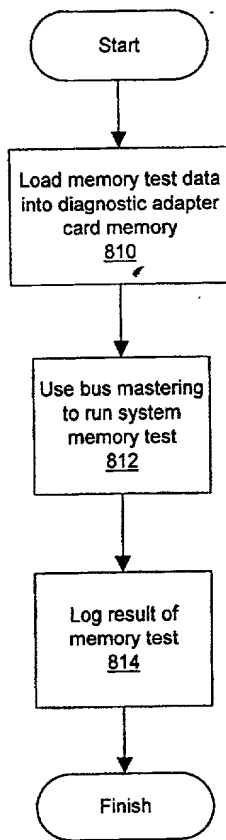
AUS000213US1



**Figure 7A**  
AUS000213US1

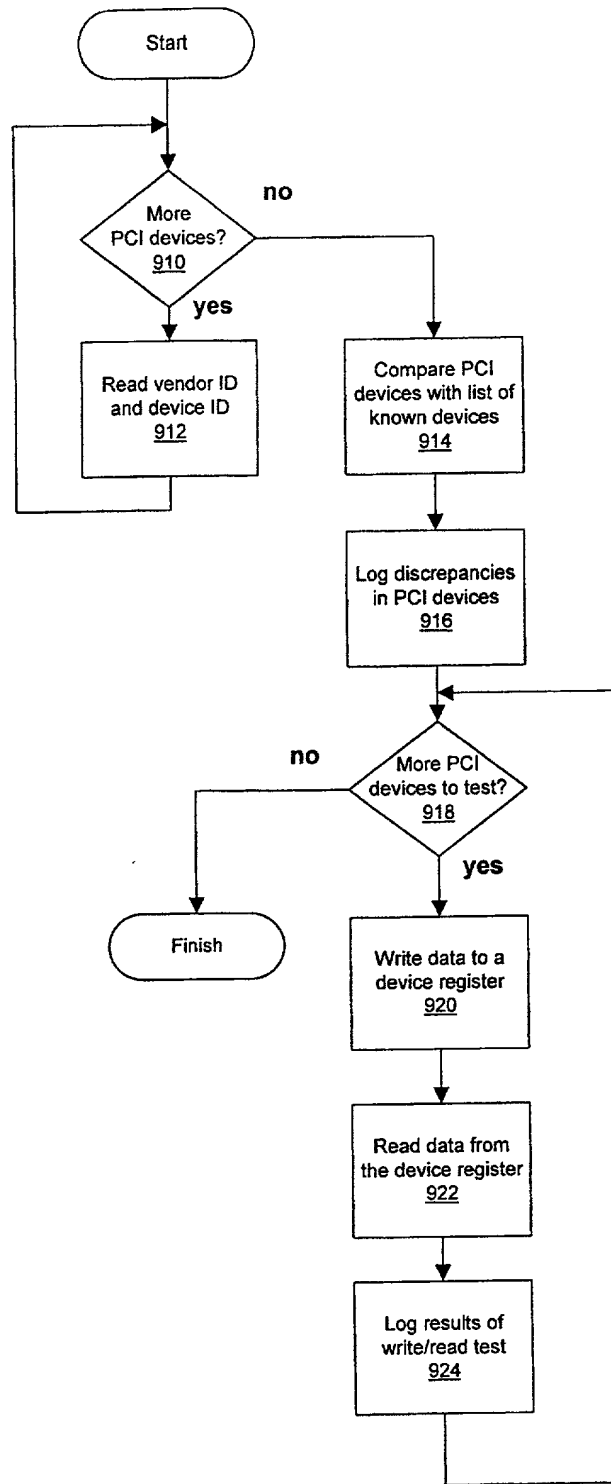


**Figure 7B**  
AUS000213US1



**Figure 8**

AUS000213US1



**Figure 9**

AUS000213US1

**DECLARATION AND POWER OF ATTORNEY FOR  
PATENT APPLICATION**

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name;

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled

**AN APPARATUS AND A METHOD FOR DIAGNOSING PROBLEMS ON A NETWORK COMPUTER**

the specification of which (check one)

X is attached hereto.

— was filed on \_\_\_\_\_  
as Application Serial No. \_\_\_\_\_  
and was amended on \_\_\_\_\_  
(if applicable)

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose information which is material to the patentability of this application in accordance with Title 37, Code of Federal Regulations, §1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, §119 of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

Prior Foreign Application(s):

Priority Claimed

\_\_\_\_ Yes \_\_\_\_ No  
(Number) (Country) (Day/Month/Year)

I hereby claim the benefit under Title 35, United States Code, §120 of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, §112, I acknowledge the duty to disclose information material to the patentability of this application as defined in Title 37, Code of Federal Regulations, §1.56 which occurred between the filing date of the prior application and the national or PCT international filing date of this application:

(Application Serial #) (Filing Date) (Status)

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

POWER OF ATTORNEY: As a named inventor, I hereby appoint the following attorneys and/or agents to prosecute this application and transact all business in the Patent and Trademark Office connected therewith.

John W. Henderson, Jr., Reg. No. 26,907; Thomas E. Tyson, Reg. No. 28,543; James H. Barksdale, Jr., Reg. No. 24,091; Casimer K. Salys, Reg. No. 28,900; Robert M. Carwell, Reg. No. 28,499; Douglas H. Lefevre, Reg. No. 26,193; Jeffrey S. LaBaw, Reg. No. 31,633; David A. Mims, Jr., Reg. No. 32,708; Volel Emile, Reg. No. 39,969; Anthony V. England, Reg. No. 35,129; Leslie A. Van Leeuwen, Reg. No. 42,196; Christopher A. Hughes, Reg. No. 26,914; Edward A. Pennington, Reg. No. 32,588; John E. Hoel, Reg. No. 26,279; Joseph C. Redmond, Jr., Reg. No. 18,753; Marilyn S. Dawkins, Reg. No. 31,140; Mark E. McBurney, Reg. No. 33,114; Duke W. Yee, Reg. No. 34,285; Colin P. Cahoon, Reg. No. 38,836; Rudolph J. Buchel, Reg. No. 43,448; Stephen R. Loe, Reg. No. 43,757; Stephen J. Walder, Jr., Reg. No. 41,534; Charles D. Stepps, Jr., Reg. No. 45,880; and Stephen R. Tkacs, Reg. No. P-46,430.

Send correspondence to: Duke W. Yee, Carstens, Yee & Cahoon, LLP, P.O. Box 802334, Dallas, Texas 75380 and direct all telephone calls to Duke W. Yee, (972) 367-2001

FULL NAME OF SOLE OR FIRST INVENTOR: James Arthur Brewer

INVENTORS SIGNATURE: James Arthur Brewer DATE: 06/21/00

RESIDENCE: 18002 North Rim  
Leander, Texas 78641

CITIZENSHIP: United States

POST OFFICE ADDRESS: SAME AS ABOVE

FULL NAME OF SECOND INVENTOR: Sanjay Gupta

INVENTORS SIGNATURE: [Signature] DATE: 06/21/00

RESIDENCE: 11900 Hobby Horse Court #312  
Austin, Texas 78758

CITIZENSHIP: India

POST OFFICE ADDRESS: SAME AS ABOVE

FULL NAME OF THIRD INVENTOR: Nicole Dawn Hartman

INVENTORS SIGNATURE: Nicole Dawn Hartman DATE: 06/21/00


RESIDENCE: 12707 Pond Woods Rd. #1815  
Austin, Texas 78729

DOCKET NUMBER: AUS000213US1

CITIZENSHIP: United States

POST OFFICE ADDRESS: SAME AS ABOVE

FULL NAME OF FOURTH INVENTOR: Mark Alan Jacks

INVENTORS SIGNATURE:  DATE: 06/21/2000

RESIDENCE: 6005 Highlandale Drive  
Austin, Texas 78731

CITIZENSHIP: United States

POST OFFICE ADDRESS: SAME AS ABOVE